CreateProcess-01

Close process and thread handles after calling CreateProcess()

Sean Barnum, Cigital, Inc. [vita[1]]

2007-03-20

Part "Original Cigital Coding Rule in XML"

Mime-type: text/xml, size: 5578 bytes

<table>
<tr><td><b>Attack Category</b></td><td>• Denial of Service</td></tr>
<tr><td><b>Vulnerability Category</b></td><td>• Process management</td></tr>
<tr><td><b>Software Context</b></td><td>• Threads and Processes</td></tr>
<tr><td><b>Location</b></td><td></td></tr>
<tr><td><b>Description</b></td><td>After calling CreateProcess(), ensure that process and thread handles get closed.
CreateProcess() and related functions create a new process. A ProcessInformation argument is returned, and this contains handles for the process and thread associated with the new process. Remember to close these process and thread handles; otherwise the system will not be able to clean up after the child process exits. The system will be able to clean up once both the child and parent processes exit, but this may tie up resources longer than necessary.</td></tr>
<tr><td><b>APIs</b></td><td><table>
<tr><th>FunctionName</th><th>Comments</th></tr>
<tr><td>CreateProcess</td><td></td></tr>
<tr><td>CreateProcessA</td><td>ANSI implementation</td></tr>
<tr><td>CreateProcessW</td><td>Unicode implementation</td></tr>
<tr><td>CreateProcessAsUser</td><td></td></tr>
<tr><td>CreateProcessWithLogonW</td><td></td></tr>
</table></td></tr>
<tr><td><b>Method of Attack</b></td><td>By repeatedly creating processes and threads that do not get cleaned up, an attacker can eventually force the system into resource exhaustion and lead to a Dos.</td></tr>
<tr><td><b>Exception Criteria</b></td><td></td></tr>
<tr><td><b>Solution Applicability</b></td><td><table>
<tr><th>Solution Applicability</th><th>Solution Description</th><th>Solution Efficacy</th></tr>
<tr><td>Whenever creating a new process.</td><td>Close thread and process handles in process</td><td>Effective.</td></tr>
</table></td></tr>
</table>

[1] http://buildsecurityin.us-cert.gov/bsi/about_us/authors/35-BSI.html (Barnum, Sean)

	information struct once they are no longer needed.

Signature Details	BOOL CreateProcess(LPCTSTR lpApplicationName, LPTSTR lpCommandLine,LPSECURITY_ATTRIBUTES lpProcessAttributes,LPSECURITY_ATTRIBUTES lpThreadAttributes,BOOL bInheritHandles,DWORD dwCreationFlags,LPVOID lpEnvironment,LPCTSTR lpCurrentDirectory,LPSTARTUPINFO lpStartupInfo, LPPROCESS_INFORMATION lpProcessInformation);
	BOOL CreateProcessAsUser(HANDLE hToken,LPCTSTR lpApplicationName,LPTSTR lpCommandLine, LPSECURITY_ATTRIBUTES lpProcessAttributes,LPSECURITY_ATTRIBUTES lpThreadAttributes,BOOL bInheritHandles,DWORD dwCreationFlags,LPVOID lpEnvironment,LPCTSTR lpCurrentDirectory, LPSTARTUPINFO lpStartupInfo,LPPROCESS_INFORMATION lpProcessInformation);
	BOOL CreateProcessWithLogonW(LPCWSTR lpUsername,LPCWSTR lpDomain,LPCWSTR lpPassword, DWORD dwLogonFlags,LPCWSTR lpApplicationName,LPWSTR lpCommandLine,DWORD wCreationFlags,LPVOID lpEnvironment,LPCWSTR lpCurrentDirectory,LPSTARTUPINFOW lpStartupInfo,LPPROCESS_INFORMATION lpProcessInfo);

Examples of Incorrect Code	`STARTUPINFO si;`
	`PROCESS_INFORMATION pi;`
	`BOOL fSuccess;`
	`fSuccess = CreateProcess(NULL, TEXT("MyProgram.exe"), NULL, NULL, TRUE, 0, NULL, NULL, &si, &pi);`
	`[...]`
	`/* Handles in "pi" not closed, preventing cleanup when MyProgram.exe exits */`

Examples of Corrected Code	`STARTUPINFO si;`
	`PROCESS_INFORMATION pi;`
	`BOOL fSuccess;`
	`fSuccess = CreateProcess(NULL, TEXT("MyProgram.exe"), NULL, NULL, TRUE, 0, NULL, NULL, &si, &pi);`
	`[...]`

```
/* Use handles if needed */
[...]
if (!CloseHandle(pi.hProcess))
{ /* handle error */ }
if (!CloseHandle(pi.hThread)) { /
* handle error */ }

/* Now resources can be released
when MyProgram.exe exits. */
```

Source Reference	- http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dllproc/base/processes_and_threads.asp[2]
Recommended Resources	- MSDN Processes and Threads[3] - MSDN CreateProcess reference[4]
Discriminant Set	**Operating System** - Windows **Languages** - C - C++

Cigital, Inc. Copyright

The Build Security In (BSI) portal is sponsored by the U.S. Department of Homeland Security (DHS), National Cyber Security Division. The Software Engineering Institute (SEI) develops and operates BSI. DHS funding supports the publishing of all site content.

[1] mailto:copyright@cigital.com